

Equivalent to Ref. 4

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2005-510805

(P2005-510805A)

(43) 公表日 平成17年4月21日(2005.4.21)

(51) Int.Cl.<sup>7</sup>

G06F 12/00

F I

G06F 12/00 533J

テーマコード(参考)

5B082

審査請求 未請求 予備審査請求 有 (全 32 頁)

(21) 出願番号	特願2003-548121 (P2003-548121)	(71) 出願人	504202944
(86) (22) 出願日	平成14年11月26日 (2002.11.26)		コグニマ リミテッド
(85) 翻訳文提出日	平成16年5月26日 (2004.5.26)		Cognima Ltd
(86) 国際出願番号	PCT/GB2002/005308		イギリス国 ロンドン W1W 5BB,
(87) 国際公開番号	W02003/046759 Reference 4		グレート ティッチフィールド ストリ
(87) 国際公開日	平成15年6月5日 (2003.6.5)		ート 131-151
(31) 優先権主張番号	0128243.3		131-151 Great Tichf
(32) 優先日	平成13年11月26日 (2001.11.26)		ield Street, London
(33) 優先権主張国	英国 (GB)		W1W 5BB United Kin
			gdom
		(74) 代理人	100076428
			弁理士 大塚 康德
		(74) 代理人	100112508
			弁理士 高柳 司郎

最終頁に続く

(54) 【発明の名称】 コンピュータ装置間でデータを複製する方法

## (57) 【要約】

資源制約型無線コンピュータ装置（例えば、携帯電話）にデータベースレコードに対する複製機能が与えられる（例えば、連絡先、電子メール、写真などをリモートサーバへとバックアップできるようにする）。この機能は、低帯域幅の信頼性の低い無線接続を使用して過度な処理負荷を負うことなく動作する。機能は、各データベースレコードにタイムスタンプを含まずに、変更ログレコードのみにタイムスタンプを押すことによって達成される。この手法では、従来行われているように各データベースレコードにタイムスタンプを押す必要がないので、相当なメモリ空間を節約することになる。変更ログは複製対象データを定義する。変更ログは単独でデータベースのマスタコピーのホストとして機能するメインサーバへと送信され、最新の状態に維持されなければならない。変更ログはコンパクトであるので、データ複製の目的で送信すべきデータは格段に少なくなり、通常、変更されたフィールドとそのフィールドが無線コンピュータ装置上で変更された方法及び日時だけである。従来のシステムでは、変更されていないデータを含むにも関わらず、通常、レコード全体を送信する。

## 【特許請求の範囲】

## 【請求項 1】

複数のコンピュータ装置における第 1 のコンピュータ装置が、無線ネットワークを介して無線コンピュータ装置から受信されたデータが複製されたか否かを判定する複数のコンピュータ装置からのデータを複製する方法であって、

(a) データベース中のフィールドエントリがいつ変更されたのかを示すためのタイムスタンプフィールドを含まない複数のレコードを伴う前記データベースを稼動させる工程と

(b) 前記データベースとは別であり且つ変更されたレコードに対してのみデータを含む変更ログにおいてタイムスタンプを生成・格納する工程と、

(c) 前記第 1 のコンピュータ装置上で稼動するデータベースにおいて前記変更ログ中のデータを複製するために、前記第 1 のコンピュータ装置へその変更ログを送信する工程とを有することを特徴とする方法。

10

## 【請求項 2】

前記変更ログは変更されたレコードのフィールドへの変更のみを記録し、前記レコードの全体を記録しないことを特徴とする請求項 1 記載の方法。

## 【請求項 3】

前記無線コンピュータ装置と前記第 1 のコンピュータ装置との間で最後のデータ複製以来発生した変更のみを記録する工程を有することを特徴とする請求項 1 記載の方法。

## 【請求項 4】

前記無線コンピュータ装置上でいつどのようにデータが変更されたかを識別するデータを前記変更ログに入れる工程を有することを特徴とする請求項 1 記載の方法。

20

## 【請求項 5】

前記無線コンピュータ装置及び前記第 1 のコンピュータ装置が前記変更ログの適切な解釈を定義するメタデータを共有する工程を有することを特徴とする請求項 1 記載の方法。

## 【請求項 6】

新規のアプリケーションに関連する変更ログが前記無線コンピュータ装置によってどのように表されるかを定義するために、前記無線コンピュータ装置が前記第 1 のコンピュータ装置に新規のメタデータ定義を登録する工程を有することを特徴とする請求項 5 記載の方法。

30

## 【請求項 7】

ユーザの介入なしに前記変更ログが前記無線コンピュータ装置によって前記第 1 のコンピュータ装置へ自動的に送信される工程を有することを特徴とする請求項 1 記載の方法。

## 【請求項 8】

無線コンピュータ装置が、(a) 所定の定期的な間隔で、(b) 所定の時刻に、或いは(c) 変更ログが生成されたときはいつでも継続的に前記変更ログを自動的に送信する工程を有することを特徴とする請求項 1 記載の方法。

## 【請求項 9】

前記無線コンピュータ装置が複数の変更ログをまとめる工程を有することを特徴とする請求項 1 記載の方法。

40

## 【請求項 10】

前記無線コンピュータ装置が、ネットワークトラフィックがより低い時間に低い優先度の変更ログと共に、前記第 1 のコンピュータ装置で迅速に処理される変更ログの重要度に基づいた時間にバッチ・変更ログを送信する工程を有することを特徴とする請求項 9 記載の方法。

## 【請求項 11】

前記無線コンピュータ装置が、ネットワークトラフィックがより低い時間に低い優先度のレコードと共に、前記第 1 のコンピュータ装置で迅速に処理されるレコードの重要度に基づいた時間に変更ログ中のレコードを送信する工程を有することを特徴とする請求項 1 記載の方法。

50

## 【請求項 1 2】

第 1 のコンピュータ装置がデータを複製するために、無線ネットワークを介して前記第 1 のコンピュータ装置へ前記データを送信するようにプログラムされた無線携帯コンピュータ装置において、

(a) データベースの中のフィールドエントリがいつ変更されたのかを示すためのタイムスタンプフィールドを含まない複数のレコードを備える前記データベースを稼働させ、

(b) 前記データベースから分離し、変更されたレコードのためだけのデータを含む変更ログにタイムスタンプを生成及び格納し、

(c) 前記第 1 のコンピュータ装置上で稼動するデータベースにおいて前記変更ログ中のデータを複製するために、前記第 1 のコンピュータ装置へこれらの変更ログを送信するようにプログラムされた装置。

10

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

本発明は、コンピュータ装置間でデータを複製する方法に関し、特に、無線ネットワークを介してサーバと同期する携帯電話及び電子手帳などの資源制約型無線装置（すなわち、処理能力及びバッテリー寿命に制限のある装置）に関する。

## 【背景技術】

## 【0002】

コンピュータ装置間でのデータ複製は、同一人物によって使用される異なるコンピュータ装置間で文書、電子メール、連絡先、及びカレンダーへの記入を同期させる際などの多様な状況において行われる。データ複製を正しく行うためには、2 つ以上の矛盾するデータ複製要求のうちの何れに従うべきかを決定する何らかのシステムが必要である。例えば、時点 1 において、ユーザが携帯電話上で人物 X の電子メールの詳細に対する変更を入力するとする。しばらくして、時点 2 において、ユーザはその変更に誤りがあったことに気づく。人物 X の電子メール詳細に更なる変更を加え、同じことを PC 上でも実施する。時点 3 において、ユーザは PC をマスタサーバに同期させて、人物 X の電子メールのマスタレコードを正しいアドレスへと更新する。しかし、更に時間が経過してから、時点 4 において、携帯電話をマスタサーバに同期させるとする。この場合、人物 X の正しい電子メール詳細（時点 2 で入力）を時点 1 で最初に入力した不正確な詳細で上書きする可能性がある。

20

30

## 【0003】

この問題に対処するために、同期化システムは、通常、最後に記録された変更（上述の例では時点 1 ではなく時点 2）を優先するなどの時間ベースの仲裁規則を使用する。これは、装置上でデータがいつ変更されたのかを示すために、タイムスタンプをそのデータと関連付けることをその装置に要求する。この処理は通常、ある特定の種類のデータを格納するコンピュータ装置上でデータベース（例えば、個人の全ての連絡先詳細一名前、電子メール、電話、携帯電話などであり、これらで単一の連絡先レコードを形成する一に対するデータベース。通常、電子メール、カレンダーへの記入などに対してはそれぞれ別のデータベースが存在する）を提供することにより、有線ベースの装置に対して行われる（無線ネットワークを介して通信する装置とは対照的に）。レコードの何らかのデータが変更されると（例えば、ある個人の新規の電子メールアドレス）、装置はその変更をレコードの一部として記録し、装置上でローカルに稼動する水晶時計により設定される時間でタイムスタンプを押す（レコードの一部を形成するエントリとして）。データの変更（例えば、その個人に対する新規の電子メール連絡先詳細）は中央サーバへと送られる。完全な置換レコードが、通常、送信される（例えば、個人に対する全ての既存の連絡先詳細＋新規の電子メールアドレス＋タイムスタンプ）。中央サーバは、情報のマスタコピーを更新し、全ての装置が同じ情報を維持するように各装置に更新情報を送信する。中央サーバは、通常、タイムスタンプを使用して仲裁規則を適用し、競合／矛盾する同期化要求間の競合を解消する。

40

50

## 【0004】

従って、従来の有線ベースのデータ複製システムは、データベースレコードに適用されるタイムスタンプに依存する。タイムスタンプは、変更されたデータが最初に入力された装置上で稼動するローカルの水晶時計により作成される。この装置は、クライアント装置であっても、サーバであっても、あるいは、ピアであっても良い。

## 【0005】

資源制約型無線装置に関しても、本質的に同じ手法を使用するが、データベースレコード自体は固定長レコードであるのが普通である。レコード間の境界を容易に判定することができるので、索引付けが促進される。しかし、これにより、変更が行われなかったことを示すためにレコードが「ヌル」エントリをタイムスタンプフィールドに含むことになるか、あるいは、マスタデータベースを保持するサーバへの同期化より前の日時であるので不必要であるにも関わらず、最終タイムスタンプを保持することになる。何れの手法も不要なメモリ空間を占有する。更に、同期化が行われると、装置上の同期化エンジンは、データベース（通常、論理シーケンス）中の全てのレコードを分析しなければならず、最終同期化時間より後のタイムスタンプを有するレコードの全体をコピーし、それをサーバに送信することになる。資源制約型無線装置上では、全てのデータベースレコードを分析する処理は時間がかかる可能性があり、貴重なプロセッサ能力を使い尽くし、電力を消耗させる。

## 【0006】

## 〔発明の概要〕

本発明の第1の態様によると、複数のコンピュータ装置における第1のコンピュータ装置が、無線ネットワークを介して無線コンピュータ装置から受信されたデータが複製されたか否かを判定する複数のコンピュータ装置からのデータを複製する方法であって、

(a) データベース中のフィールドエントリがいつ変更されたのかを示すためのタイムスタンプフィールドを含まない複数のレコードを伴う前記データベースを稼動させる工程と

(b) 前記データベースとは別であり且つ変更されたレコードに対してのみデータを含む変更ログにおいてタイムスタンプを生成・格納する工程と、

(c) 前記第1のコンピュータ装置上で稼動するデータベースにおいて前記変更ログ中のデータを複製するために、前記第1のコンピュータ装置へその変更ログを送信する工程とを有することを特徴とする方法が提供される。

## 【0007】

この手法により、資源制約型無線携帯コンピュータ装置（例えば、携帯電話）は、低帯域幅の信頼度の低い無線接続を使用して過剰な処理負荷を負うことなくデータ複製機能（例えば、連絡先、電子メール、写真などのデータベースをリモート装置—データベースのマスタコピーを格納するサーバ又はピア—へとバックアップする）を提供できるようになる。機能は、変更されたデータベースから変更ログを切り離すことによってこれを実施する。リモート装置へと送信すべきデータは格段に少なくなり、通常、変更されたフィールドとそのフィールドが無線コンピュータ装置上で変更された方法及び日時だけである。従来のシステムでは、変更されていないデータを含んでいる時でさえ、通常、レコード全体を送信する。更に、データベース自体にタイムスタンプを含まずに、変更ログレコードにのみタイムスタンプを押すことによって、この手法では、従来行われているように各データベースレコードにタイムスタンプを押す必要がないので、無線装置上で相当なメモリ空間を節約することになる。通常、変更ログは、無線コンピュータ装置と第1のコンピュータ装置との間の最後のデータ複製以降発生した変更のみを記録する。従来のシステムでは、同期化より前のタイムスタンプを記録することによって貴重なメモリを浪費することが多い。

## 【0008】

前記無線コンピュータ装置及び前記第1のコンピュータ装置（携帯装置とは限らず、むしろ、大型のサーバの方が普通）は前記変更ログの適切な解釈を定義するメタデータを共

10

20

30

40

50

有する。前記無線コンピュータ装置は、前記第1のコンピュータ装置を用いて新規のメタデータ定義を登録し、どのように新規のアプリケーションに関連する変更ログが前記無線コンピュータ装置により提示されるかを定義しても良い。

【0009】

別の特徴として、前記変更ログは、ユーザ介入なしに前記無線コンピュータ装置により前記第1のコンピュータ装置へと自動的に送信される。この送信は、(a) 所定の定期的な間隔で、(b) 所定の時刻に、或いは(c) 変更ログが生成されたときはいつでも継続的に行っても良い。何れにしても、第1のコンピュータ装置上のマスタデータベースは最新の状態に維持されるので、ユーザが幾つかの異なる装置から同じデータを手動で同期させる際に起こり得る問題(従来技術の欄で説明したような)の発生は格段に減少する。

10

【0010】

複数の変更ログはまとめることができる。まとめられた変更ログは、前記第1のコンピュータ装置により迅速に処理される変更ログの重要度によって異なる時間に送信し、ネットワークトラフィックがより低い時間に低い優先度の変更ログを送信しても良い(これにより請求金額が抑えられる)。同様に、変更ログ中のレコードは、前記第1のコンピュータ装置により迅速に処理されるレコードの重要度によって異なる時間に送信し、ネットワークトラフィックがより低い時間に低い優先度のレコードを送信することができる。

【0011】

第2の態様において、第1のコンピュータ装置がデータを複製するために、無線ネットワークを介して前記第1のコンピュータ装置へ前記データを送信するようにプログラムされた無線携帯コンピュータ装置において、

20

(a) データベースの中のフィールドエントリがいつ変更されたのかを示すためのタイムスタンプフィールドを含まない複数のレコードを備える前記データベースを稼働させ、

(b) 前記データベースから分離し、変更されたレコードのためだけのデータを含む変更ログにタイムスタンプを生成及び格納し、

(c) 前記第1のコンピュータ装置上で稼働するデータベースにおいて前記変更ログ中のデータを複製するために、前記第1のコンピュータ装置へこれらの変更ログを送信するようにプログラムされた装置が提供される。

【発明を実施するための最良の形態】

【0012】

30

本発明は、Cognima Limited of London, United Kingdomによる実施例を参照して説明する。コグニマ(Cognima)では、消費者によるデータサービスの利用を増加させ、大切な顧客の固定客化を促進し、競合他社から自社のサービスを差別化するために、モバイルサービスプロバイダ(MSP)及びネットワークオペレータの必要性に直接対処するデータ複製技術を開発してきた。

【0013】

コグニマのデータ複製ソリューションは、これらの問題に対して以下の手法により対処する：

- ・注目度の高い、容易に利用可能なものにすることにより、データサービスの利用を増加すること

40

- ・加入者の個人データを担当のMSPにより制御されたサーバ上で確実にバックアップすることにより、乱れ(churn)の防止を強化すること

- ・顧客の経験を制御することによってMSPが差別化されたサービスを作り出せるようにすること。

【0014】

1. コグニマデータ複製フレームワークの用途の概説

コグニマのデータ複製フレームワークにより、モバイルサービスプロバイダは消費者市場に対して注目度の高いサービスを設立することができる。MSPは、データセンタにおけるコグニマサーバのホストとして機能する。サーバは、規格ベースのJ2EEアプリケーションサーバ及びキャリア級のUnixハードウェア上で動作するオラクルデータベ

50

ス及びコグニマのマルチスレッドＪａｖａ通信サーバから構成される。第４節以降では、技術的な実現を詳細に説明する。

【００１５】

コグニマフレームワークは、携帯電話に入力されたデータを自動的に（ユーザの介入なしに）コグニマサーバを介して他の電話機へと複製する。同様に、コグニマサーバに接続された外部システムからのデータは、携帯電話上で自動的に最新の状態に維持される。

【００１６】

コグニマ対応アプリケーションを使用するモバイル加入者は、「個人情報」及び「複数の友人」に対する「常時利用可能な」、「即時」接続を経験する。

・「個人情報」は、加入者の住所録、メッセージ、銀行口座取引明細、株価、ピザ注文、カレンダー、職場への通勤経路の現在の交通状況、又はその他の個人化されたコンテンツを含むことができる。データは常にコグニマサーバ上で確実にバックアップされ、全ての関連するクライアント装置上で自動的に複製される

・「常時利用可能な」は、現在ネットワークに接続されているか否かに関わらず、加入者が携帯する何れの装置又はハンドセット上でも個人情報をアクセス可能であることを意味する。ユーザは、装置上にローカルに格納された個人情報を常にアクセスすることができるからである。また、ユーザはウェブインタフェース－Virtual Phone－を介してサーバ上で直接個人情報を編集・管理することができる

・「即時」は、加入者がサーバからのデータのダウンロードを待つ必要がないことを意味する。最新の情報は、加入者がその必要性を認識していなくても既にそのハンドセット上にある。データが自動的にハンドセットへと送信されるからである（例えば、ハンドセットによるポーリングが行われるであろう。このポーリングは、例えば３０分ごと又は所定の時刻（午後４時、午後５時など）に定期的に行うことができる。ハンドセットに対するプッシングも起こり得る）

・加入者は複数の装置に渡って彼らの「複数の友人」とデータを共有することができる。コグニマサーバはこのデータを任意の定義済の装置又は定義済の個人へと複製することができるからである。

【００１７】

１．１ コグニマアプリケーションの例

顧客	ニーズ	コグニマアプリケーション
<b>Sarah</b>	Sarah は、重要な連絡先の番号及びメッセージを含む電話機を盗まれた。彼女は、手動ではバックアップコピーを取っていない	Sarah が電話機にデータを入力するたびに、コグニマは自動的にMSPのデータセンターの中央サーバにそれをバックアップする。Sarah は、同じMSPに所属する限りは、新しい携帯電話を購入して全ての連絡先及びメッセージを中央サーバから即時に取得することができる。また、MSPのポータルを介して盗まれた電話機からデータを削除することもできる
<b>Jill</b>	Jill は外でショッピング中である。高額の買い物をする前に、給料が銀行口座に振り込まれているか否かを知る必要がある。しかし、彼女の現在の所在地はデパートの地下であり、ネットワークの有効範囲内ではない	コグニマは、定期的に（又は所定の時刻にあるいは変化があり次第）変更されたデータを Jill の携帯電話に送信することによって、携帯電話上で Jill の個人化されたコンテンツ（銀行口座取引明細を含む）を自動的に最新の状態に維持する。最新のデータは、Jill がその必要性を認識していなくても既に彼女の電話上にある。ネットワークの有効範囲内でも、そのデータを即時にアクセスすることができる
<b>Matthew</b>	Matthew は、自身の現在の空き状況及び「気分」を友人に常に知らせておきたい。また、友人が何をしているのかを知りたい。Matthew は、主に、交際する仲間内での出来事を常に把握しておくことに興味があり、「オンライン」状態に移行したり、多くの費用のかかるメッセージを送信したりすることなく、一目で把握したいと思っている	コグニマは Matthew のプレゼンスプロフィールを彼の友人と共有する。Matthew が自身のプロフィールを変更する（例えば、彼が社交的な気分であることを示すアイコンを選択する）場合、アイコンは自動的に彼の住所録において友人の電話機上のエントリを更新する。Matthew は、自身の電話機上で一見して全ての友人のプレゼンス情報を見ることができる。彼は、友人が社交的な気分になったり退屈に感じたりした時にすぐに電話をかけることができるように、電話機に通知を要求することもできる
<b>Laura</b>	Laura は2台の携帯電話を持っている。1台は職場で使用し、もう1台は夜持ち歩くためのファッションフォンである。Laura は双方の電話機で同じ住所録を維持した	コグニマは Laura の各電話機の全てのデータを同期させる。Laura が1台のハンドセットでデータを編集すると、即時に（又は定期的あるいは所定の時刻に）コ

10

20

30

40

## 2. モバイル加入者にとっての利点

コグニマは、以下の重要な利点に基づいて量販市場の消費者データサービスを実現するための理想的なフレームワークを提供する：

・「親しみやすさ」：ユーザの介入は不要である。加入者は自身のデータをアクセスするのに「同期」又は「ダウンロード」ボタンを押下する必要がある。システム構成及び確実

50

なデータ転送は、エンドユーザからは完全に見えない

・「即時可用性」：ユーザは常に、ローカルデータとは即時に相互に作用することができる（オフライン時であっても）。一方で、更新はバックグラウンドでひそかに行われる。例えば、ユーザは地下鉄でも個人化されたコンテンツを読むことができる。ユーザ行為（user experience）はデータ転送から切り離されている

・「値ごろ感」：MSPは複製をいつ実行するかを制御でき、もたらされるサービス品質（QoS）を制御できる。しかし、ユーザ行為はデータ転送から切り離されているので、低いQoSでもユーザのサービスに対する認識には影響しない。このため、MSPはユーザ行為を犠牲にすることなく、比較的低いQoSで低コストな加入ベースのサービスを提供することができる。例えば、銀行口座取引明細などの緊急でないデータサービスに関しては、データ複製を夜間に行うことも可能であり、この場合でもユーザは不満を感じない。夜間のデータ複製では、十分に利用されていない帯域幅を使用するため、ピーク時のデータ複製に比べて格段に費用が安い。緊急のデータ複製（例えば、プレゼンス情報）は、定期的に又は（オプションとして）継続的に（プッシュで）随時行うことができ、請求料金をより高額なものにすることができる。更に、電話機のメモリ及びプロセッサ能力の有効利用により、コグニマクライアントソフトウェアは、安価な量販市場の電話機においても、経済的にインストールすることができる。

10

#### 【0018】

##### 3. モバイルサービスプロバイダにとっての利点

コグニマは、新規のデータ収入を生み出し、乱れを減少させ、自社のサービスを競合他社のサービスから差別化する手段を提示する。

20

#### 【0019】

##### 3. 1 既存のモバイルサービスの利用度の向上

コグニマは、既存のモバイルサービスの利用度を向上する：

- ・メッセージングサービス及びコンテンツベースのサービスは、ますます便利になると共に即時性が高まり、両サービスの利用度が向上することになる
- ・プレゼンス情報の即時性が向上するため、チャット、インスタントメッセージング及び音声通話の空き容量があることを示すアラートの利用が増加することになる
- ・複数の装置を容易に管理することができるので、ユーザはいかなる場合でも適切な電話機を携帯することができ、通話やメッセージ送信が増加する。

30

#### 【0020】

##### 3. 2 注目度の高い新規のサービス

コグニマは、注目度が高くお手頃な新規のモバイルデータサービスの急速な導入を可能にする。

・コグニマは、余剰のネットワーク容量のみを使用し、ローエンドの電話機における新規のサービスに対する注目度の高いユーザ行為をもたらす。これは、ネットワークオペレータにとって入手可能であると共にスケーラブルであり、MSPは量販市場の加入者に対して理解可能且つ予測可能な値段設定を提供することができる

・新規のコグニマサービスに対するアプリケーション開発の大半は、サーバ側で行われるため、MSPは新規のサービスを市場に迅速にもたらしすることができる

40

・コグニマのクライアントソフトウェアは、フラッシュメモリのアップグレードとしてインストールすることができ、今日の量販市場のハンドセットに多機能電話並みの機能を付与する。新規のソフトウェアアプリケーションは、無線により既存のコグニマ対応ハンドセットへとダウンロードすることができるので、MSPは新規のデータサービスをそのサービスに対応した新規の装置を待たずとも展開することができる

・サードパーティのアプリケーション開発者は、MSPのコグニマインフラストラクチャを活用して、MSPのネットワークに対する新規のアプリケーションを開発することができる。

#### 【0021】

##### 3. 3 乱れ削減

50



コグニマは、乱れを防止する重大な障壁となる。例えば、個人情報にMSPのコグニマサーバに確実に格納する加入者は、新しい電話機を購入し、全ての個人情報を即時に新規の装置へと取得することができる。異なるサービスプロバイダに加入することにした場合、この個人情報は全て失われることになるだろう。

#### 【0022】

##### 3.4 差別化

現在加入者は、全てのネットワーク上でモバイルデータサービスを使用するという同じ基本的な経験を有する。例えば、WAPサービスを使用する経験は、WAPプロトコル、電話機中のブラウザ、及びアクセスされるコンテンツにより定義される。多くのMSPは、独自のユーザ行為を加入者に与えることで自身の差別化を図らなければならないことを認識している。しかし、モバイルハンドセットにおいてサービスをカスタマイズすることに対する厳しい制約が、その差別化の障害となっている。

10

#### 【0023】

コグニマにより、MSPはハンドセット上でサービスを実現し、加入者のユーザ行為の制御を取り戻すことができるようになる。最も重要なことは、コグニマは相互運用性を犠牲にすることなく、これを実現するということであり、コグニマサーバとの簡単な統合を介して業界標準への対応も達成される。最終的には、価値連関表におけるMSPの地位は、ハンドセットメーカー及びコンテンツプロバイダの強力なブランドに対して強化される。

#### 【0024】

##### 4. コグニマデータ複製フレームワークの機能設計

##### 4.1 概論

詳細な説明の本節以降の各節では、コグニマデータ複製システムが実際にどのように機能するかを説明することを意図する。この説明には、クライアント装置、コグニマサーバ、及びウェブクライアントの挙動が含まれるが、特定のハードウェア、プログラミング言語、ソフトウェアクラス設計、又は環境の詳細については触れない。使用される基本的なデータ構造及びアルゴリズムについては説明する。

20

#### 【0025】

##### 用語

クライアント装置	コグニマクライアントソフトウェアを実行する電話機、PDA、又はその他の機械
コグニマサーバ	コグニマサーバソフトウェアを実行してデータを複製するクライアント装置によりアクセス可能なサーバ
複製	クライアント装置からコグニマサーバ、更に同じユーザに所属するその他のクライアント装置に至るまでデータをコピーするプロセス
ユーザ	少なくとも1台のコグニマクライアント装置を所有し、使用する者
ユーザデータ	ユーザがクライアント装置上に格納し、操作することを希望するであろう1組の情報（連絡先、メッセージ、呼出し音、画像など）

30

40

##### 4.2 目的

コグニマソフトウェアの目的は以下の通りである：

- ・ユーザ即時アクセスにより、無線データ接続が可能な複数のハンドヘルド装置上でデータの「最新の」コピーを表示・修正できるようにする
- ・ユーザが従来のウェブブラウザを使用して同じデータを表示・修正できるようにする
- ・ユーザのデータの確実なバックアップを簡単に実現できるようにする
- ・複雑で費用のかかる処理をサーバに転嫁することによって、安価なハンドセット上で強力なデータ機能をユーザに対して提供する。

50

## 【0026】

## 4. 3 最高レベルの説明

クライアント装置は、クライアント装置上のデータベースにユーザのデータのコピーを保持する。ネットワーク接続の有無に関わらずこのデータにアクセスすることができるので、ユーザは常に即時アクセス権を有することになる。ユーザが装置上でデータを変更すると、変更は変更ログへとコピーされる。クライアント装置は、定期的に無線ネットワーク上のコグニマサーバへと接続し、変更ログから変更を送信し、新規のデータを受信する。これにより、データを変更する操作においてネットワークへと接続する必要はなくなる（すなわち、プッシュは好適な実施例では連続的ではない）。コグニマサーバは、クライアント装置から受信したデータ変更でデータベースを更新し、ユーザが所有する全ての装置の変更ログにデータ変更を反映させる。これらの装置が次に接続するときには、変更を受信し、同じデータのコピーを伴って同期状態に維持されるであろう。

10

## 【0027】

コグニマサーバはウェブサーバを含む。このウェブサーバにより、ユーザはコグニマサーバのデータベースに保持されるデータのコピーを直接調べることができ、クライアント装置上で行うように変更を加えることができる。また、コグニマサーバは、ユーザがネットワーク／インターネット上の他のサーバと通信するためのゲートウェイとしても機能する。例えば、クライアント装置はメッセージオブジェクトに少数のフラグを設定することによって、SMS、電子メール、又はファックスとしてメッセージを送信するようにコグニマサーバに事実上要求することができ、コグニマサーバは電子メールサーバ、SMSサーバ、及びファックス装置と通信を行う機能を含む。これは、呼出し音、バンキング明細、ゲームなどを保持するサーバへと拡大することができる。クライアント装置上にソフトウェアを構築するよりも、コグニマサーバ上にソフトウェアを構築して他のサーバに問い合わせる方がより簡単で安価である。

20

## 【0028】

## 5. より低いレベルの概念

## 5. 1 データ構造

## 5. 1. 1 ID

コグニマユーザデータは、オブジェクトデータベースの専門用語：クラス及びオブジェクトを用いて記述される。同様の名称を有する○○プログラミング概念と混同される可能性があるので、注意が必要である。

30

## 【0029】

コグニマネットワークの全てのユーザには「ユーザID」が割り当てられる。このIDは、コグニマネットワーク特有のものであり、ネットワークオペレータにより提供される。全てのユーザは、ユーザIDとコグニマサーバのURLとの組み合わせである「コグニマアドレス」を有する。これは世界でただ1つのものである。ユーザに所属する各装置には「装置ID」が割り当てられる。この装置IDはユーザ特有のものである。このIDは8ビットであるので、ユーザは最大253台の装置を持つことができる（ID254はウェブのために予約されており、ID255は予備であり、ID0は無効である）。全てのユーザデータは、クラス（連絡先クラス、メッセージクラス、銀行取引クラスなど）に分類され、各クラスには世界でただ1つのクラスIDが割り当てられる。例えば、クラスID「12」は連絡先のことを指す。

40

## 【0030】

クラスのインスタンスはオブジェクトであり、オブジェクトにはユーザ特有の「オブジェクトID」が割り当てられる。例えば、ある連絡先クラスオブジェクトは、「John Smith」の連絡先であるだろう。オブジェクトIDは、オブジェクトを作成した装置の装置IDを装置の寿命を超えて増加する単調増加のカウントと結びつけることにより生成される。各装置は最大16, 777, 215個のオブジェクトを作成することができる（この制限値に到達した場合には装置IDをリセットすることができるだろう）。クラスはそれを構成するプロパティにより定義される。クラスは本質的にプロパティの配列である。クラ

50

スの各プロパティはプロパティIDを持ち、このIDはクラス特有のものである（実際は、0から始まるプロパティ配列におけるプロパティの配列位置）。

#### 【0031】

##### 5.1.2 オブジェクト作成

オブジェクトは装置上で作成される。このオブジェクトにはオブジェクトIDが割り当てられ、装置データベースに保存される。コピーも変更ログへと保存される。装置が次にコグニマサーバへと接続すると、変更ログのエントリが送信される。コグニマサーバはオブジェクトをデータベースに保存し（システム時間を記録）、必要とされるクラス特有の処理（例えば、電子メールの生成・送信）を行い、ユーザが所有すると共にクラスへの関わりを宣言したその他の装置の変更ログにエントリを追加する。（このエントリは、装置上のクラスの正しいバージョンに対するものでなければならない）。

10

#### 【0032】

オブジェクトはウェブポータル上に作成されても良い。オブジェクトIDが生成され（上述のように装置ID254を使用して）、装置と同様に処理される。ウェブポータルに対しては変更ログはなく、コグニマサーバのデータベースから直接に選択内容を取得する。

#### 【0033】

また、オブジェクトはサーバアプリケーションにより作成されても良い（例えば、メッセージングモジュールは電子メールを受信し、この電子メールよりメッセージオブジェクトを作成する）。オブジェクトIDは、（上述のように装置ID254を使用して）生成され、装置と同様に処理される。

20

#### 【0034】

##### 5.1.3 オブジェクトの更新

既存のオブジェクトの1つ以上のプロパティが装置上で修正される。変更は装置データベースへと保存される。変更された各プロパティは装置変更ログにおいてエントリを生成するのに使用される。これらは、コグニマサーバへと送信される。

#### 【0035】

更新時間がコグニマサーバのデータベースのプロパティに対する「最終変更」時間より後である場合、コグニマサーバは変更をデータベースに保存し（プロパティに対する新規の「最終変更」時間を記録）、所要のクラス特有の処理を行う。更に、コグニマサーバは、ユーザに所属し、クラスを宣言すると共に、プロパティを含むクラスのバージョンを有するその他の装置の変更ログにエントリを追加する。更新は、変更が最初に行われた装置の変更ログに対しても行われる。奇妙に思えるかもしれないが、これは以下のシナリオに対処するのに必要である：

30

ユーザは2台の装置A及びBを持っている。ユーザは、午後5時にAにおいてオフラインでプロパティ7を更新し、午後6時にBにおいてオフラインでプロパティを更新する。まず、Aを用いてネットワークへと接続する。A上の値7は、Bに送信されるべく変更ログに反映される。その後、Bを用いて接続する。値7はより最近のものであるため、B上の値7がAへと送信される。しかし、BはAの値を入手する。値7をBに複製し直すことによりこれが確定される。

40

#### 【0036】

削除済としてマーク付けされたオブジェクトに対して更新がコグニマサーバにより受信され、更新が削除よりも後に行われている場合、復旧と解釈される。オブジェクトは復旧され、更新され、オブジェクトのリフレッシュが全ての適切な装置の変更ログ上で行われる。ウェブポータル又はサーバアプリケーションからの更新も同様に機能する。

#### 【0037】

##### 5.1.4 オブジェクトの削除

オブジェクトが装置上で削除される。装置データベースから削除され、クラスID及びオブジェクトIDを列挙する変更ログ上にエントリが設けられる。エントリはコグニマサーバへと送信される。

50

## 【0038】

削除時間がオブジェクトの最終更新時間より後である場合、コグニマサーバはデータベースにおいてそのオブジェクトに削除済としてマーク付けし、クラス特有の処理を行い、ユーザに所属すると共にクラスを宣言したその他の装置にエントリを追加する。

## 【0039】

削除時間が最終更新時間より前の場合、削除が無効であることを示す。オブジェクトのリフレッシュが、削除が最初に行われた装置の変更ログに対して行われる。

## 【0040】

削除されたオブジェクトは、削除状態が明らかになるようにウェブポータルで見ることができる。ユーザは復旧のためにオブジェクトを選択することができる。削除マークがコグニマサーバのデータベース中のオブジェクトから削除され、オブジェクトをリフレッシュするためのエントリが、ユーザに所属すると共にクラスを宣言した全ての装置の変更ログ上に設けられる。

## 【0041】

## 5. 1. 5 プロパティ型

各プロパティには型がある。現在、9個の許容されたプロパティ型がある：

型名	型値	型の説明
KcogTypeRef	0	別のオブジェクトの4バイトオブジェクトID
KcogTypeInt	1	符号付き4バイト整数値
KcogTypeUInt	2	符号なし4バイト整数値
KcogTypeFloat	3	符号付き4バイト浮動値
KcogTypeStr	4	CogString (文字列中の文字数を保持する4バイト符号なし整数。文字バイトが後続する)
KcogTypeTime	5	1990年1月1日の真夜中からの経過秒数を示す符号なし4バイト整数値
KcogTypeTypedStr	6	CogString が後続する符号なし4バイト整数値
KcogTypeBlob	7	バイト数を保持する4バイト符号なし整数が先行するバイトストリーム
KcogTypeArray	8	あらゆる種類のデータの配列を保持することが可能なプロブ構造

「CogString」は文字カウントであり、文字が後続する。文字列がASCIIの場合、文字列が占める空間は(4 + 文字カウント)バイトとなる。文字列がユニコードの場合、占める空間は(4 + (文字カウント \* 2))バイトとなる。

## 【0042】

「CogTypedString」は、型(4バイト符号なし整数)が先行する「CogString」である。現時点では型付き文字列の唯一の用途は接点である。型は接点の型(例えば、電子メールアドレス、自宅の電話)を識別し、文字列はアドレスを保持する(例えば、bob@xxx.yy、01233556677)。

## 【0043】

「CogBlob」はバイト長であり、バイト数が後続する。任意のバイナリデータを格納するのに使用することができる。

## 【0044】

「CogArray」は、4バイト符号なし整数「型」として渡され、2個のプロブが後続する。「型」は配列に保持される要素の型を示す。第1のプロブはインデックスプロブであり、第2のプロブへのオフセットのシーケンス(4バイト符号なし整数)を保持する。第2のプロブはデータプロブであり、バイナリランプのシーケンスとして配列の要素を保持する。各要素は、インデックスプロブに沿ってカウントし、データプロブ中の要素の開始の

オフセットを取得することによってデータプロブから抽出することができる。これが、渡されるとき「CogArray」のストリーム構造である。特定のシステムの内部では、従来のベクタ（すなわち、解析済）のように見えるかもしれない。

#### 【0045】

「CogArray」の唯一の実施例は、「MessageAddress」である。「MessageAddress」の各要素は「AddressPair」である。「AddressPair」は連絡先ID（連絡先オブジェクトのオブジェクトID）であり、接点が後続する。

#### 【0046】

##### 5. 1. 6 スマートプロパティパラメータ

プロパティの一部は「スマート」にすることができる。これは、装置の特徴に対するプロパティにデータを刻むために、一部のプロパティを特定の装置に対してパラメータ化できることを意味する。実際、パラメータは2つの4バイト符号なし整数である。一方はスマート型であり、他方は最大サイズである。例えば、メッセージの本文テキストを保持するプロパティは、メモリ容量の限られた安価な電話機では、スマート型kCogPlainTextで最大サイズ100にパラメータ化されるかもしれないが、メモリ容量の多いPDA上では、スマート型kCogRichTextで最大サイズ1,000にパラメータ化されるかもしれない。

#### 【0047】

パラメータは、アプリケーションが装置に追加されるときにコグニマサーバにより格納される。新規のオブジェクト又はそのクラスに対する更新が、その装置に対するコグニマサーバ変更ログに反映されると、スマートパラメータに従って処理される。これは、例えば、テキストを切り詰めたり、ユニコードテキストをバイト数の少ないテキストへと変換したり、あるいは、画像形式を変換したりすることを含むであろう。

#### 【0048】

データ保全の点では、コグニマサーバのデータベースで保持されるオブジェクトが生成されたときのオブジェクトのコピーであることが重要である。縮小バージョンを装置上で見る場合でも、コグニマサーバ上では完全バージョンを効果的に操作することができる。

#### 【0049】

##### 5. 1. 7 クラスバージョン

4バイト符号なし整数により定義されるクラスバージョンの概念がある。新規のクラスバージョンは古いクラスの終わりにプロパティを追加しても良いが、既存のプロパティを変更／削除したり、既存のプロパティ間に新規のプロパティを挿入したりしてはならない。バージョン間の相互運用性が可能でなければならない。異なるスマートプロパティパラメータを有するクラス定義は異なるバージョンではない。

#### 【0050】

##### 5. 2 ユーザデータ渡し

コグニマは、クラスメタデータの考えを利用してデータベース間でコピーして渡される必要のあるデータを最小限にする。クラスメタデータは、本質的にプロパティメタデータの配列である。プロパティメタデータは、プロパティID、プロパティ型、スマート型、及び最大サイズである。

#### 【0051】

ユーザデータは、クラスID以外のフォーマット情報を伴わないストリームとして転送される。このストリームは、クラスメタデータを参照することによって解析される。このため、ストリームがクラス6に対して受信され、クラス6に対するクラスメタデータが、プロパティ0がKcogTypeUIntであり、プロパティ1がKcogTypeStrであると示す場合、ストリームの最初の4バイトは符号なし整数として、次の4バイトは後続する文字列の文字数を保持する符号なし整数として、次のn（ユニコードの場合は2倍）バイトは文字列中の文字を保持するものとして解釈すべきであることが分かる。

#### 【0052】

クライアント装置は、コグニマサーバに対してサポートするクラスを宣言する。これにより、装置は生のユーザデータ（クラスID、オブジェクトID、及び少数のその他の項

10

20

30

40

50

目を含むヘッダを伴う)のみを送信し、所要帯域幅を最小限にすることができる。これは、例えば、格段に広い帯域幅を必要とするXML依存システムと対照させることができる。

#### 【0053】

コグニマサーバが装置に対するデータを刻むことができるように、クライアント装置のクラス宣言はスマートプロパティパラメータを含む。プロパティの意味がアプリケーションへとハードコードされることを強調しておいても良いだろう。クラスメタデータは、クラス7におけるプロパティ2は最大長が30文字の文字列であることを記述する。これは、クラス7におけるプロパティ2をフットボールチームの名称として解釈するアプリケーション中のコードである。

#### 【0054】

##### 5.2.1 データ複製の問題の詳細

データは、クライアント装置及びこれらの装置が接続されるサーバ（コグニマサーバとして既知）上で作成されるオブジェクトにおいて保持される。これらのオブジェクト及びこれに加えられる変更は、クライアント装置とコグニマサーバとの間で複製される。複製プロセスの設計により以下のことが可能になる：

- ・コグニマサーバ及びあるユーザに対してサーバにログオンされる全てのクライアント装置上で保持されるように複製されることになる1組のオブジェクトが定義される。任意の装置又はサーバ上で作成される新規のオブジェクトは、他の全ての装置へと複製されることになる。オブジェクトの何れかのプロパティにおける変更は、全ての装置へと複製される
- ・データにおける変更のみがクライアントからコグニマサーバへ、あるいは、その逆に送信されるため、ある更新に対して最小限のデータのみがネットワークを介して送信される
- ・設計の重要な部分は、クライアント装置上でオブジェクトの各プロパティに対して修正時間が維持される必要がないことである。制限されたクライアント装置上で修正時間を更新するのには時間がかかり、オブジェクト中の各プロパティに対する最終修正時間を維持するのには広い空間を必要とする
- ・コグニマサーバは十分な記憶空間を有し、対処するだけの処理能力を有するため、コグニマサーバ上で、オブジェクトの全てのプロパティに対する修正時間を格納しても問題はない

##### 5.2.2 メタデータ

システムが機能するためには、あるクラスのオブジェクトに対してどのプロパティが定義されるのかに関して明確に理解する必要がある。これは、クラスメタデータの定義を可能にする数少ないC++コンパイラマクロをプログラマに提供することにより行われる。

#### 【0055】

クラスにおいて使用されるプロパティの定義は、結果としてクラスメタデータ定義となる。この定義により、CRE（コグニマ認識エンジン）はプロパティがどの型であるかを認識し、データリンクを介して送受信するためにオブジェクト又はプロパティを圧縮／復元できるようになる。CREシステムが機能するためには、全てのクライアント及びサーバは同じクラスメタデータ定義を有する必要がある。従って、以下の状況が発生する：

- ・新規のメタデータ定義がクライアント装置上で宣言されると、コグニマサーバへと送信される。コグニマサーバは他の全てのクライアントに対してそのメタデータ定義を送信する
- ・新規のメタデータ定義がコグニマサーバ上で宣言されると、定義は全てのクライアント装置へと送信される
- ・新規のクライアント装置が初めてコグニマサーバへとログオンすると、何れかのオブジェクトが送信される前に全てのメタデータ定義がその装置へと送信される
- ・上記の場合の全てにおいて、コグニマサーバが、メタデータが指し示すクラス（及び特定のプロパティ）をアクセスするクライアントだけにメタデータ定義を送信するように今後は最適化されるであろう

10

20

30

40

50

### 5. 2. 3 変更ログ

変更ログの目的は、クライアント装置がコグニマサーバに（又はコグニマサーバがクライアント装置に）最後に接続した時以降に発生した変更を記録することである。コグニマAPIを使用して、アプリケーションはCREへと接続し、オブジェクトを作成／削除したり、オブジェクト中のプロパティを変更したりする。これらの変更は、変更が行われたときに変更が行われた時間と共にローカル装置上の変更ログへと追加される。常に識別することができるように、オブジェクトにはその作成時に固有の識別子が与えられる。

#### 【0056】

同様に、コグニマサーバ上で稼動するアプリケーションによるオブジェクトの作成・削除及びオブジェクトプロパティへの変更により、結果として、コグニマサーバ上でそのユーザに対して登録された全てのクライアント装置の全ての変更ログへと変更が追加されることになる。変更時間は、オブジェクトごと又はプロパティごとに記録される。

#### 【0057】

変更ログは以下の2通りの方法で作成することができる：

- ・新規のオブジェクトが作成され、プロパティが変更されたとき（これはクライアント装置においてはよくあることである）
- ・あるいは、オブジェクト及びプロパティの最終修正時間がシステム上に格納される場合、これらを使用することによって必要に応じて新規のオブジェクトを作成することができる（状況によっては、この方法は上述の方法ではなく、コグニマサーバ上で使用されることもある）

### 5. 2. 4 複製

クライアント装置が送信すべき項目を変更ログに有している場合、コグニマサーバへと接続することになる（コグニマサーバがクライアント装置へと接続する場合も同様）。デフォルトでは、変更ログの各項目は変更ログに追加された順序で送信される。しかし、特別なサービス及び緊急のデータなどの準備の際には、送信の直前に再度優先順位付けされても良い。転送される項目は以下の通りである：

- ・あるクラスのオブジェクトの各プロパティの型を含むメタデータ定義
- ・作成された新規のオブジェクト—そのオブジェクトのプロパティの内容を伴う
- ・プロパティが変更された—そのプロパティの新規の値を伴う
- ・オブジェクトが削除された

上述の何れの場合にも、関係するオブジェクト、クラス、及びプロパティを識別するために適切なIDが送信される。全ての変更ログ項目は、変更ログに追加された時間でマーク付けされる。この時間は、常に、ローカルマシン時間であり、第6. 2節で説明される時間管理手法によりGMTへと変換される。

#### 【0058】

クライアント装置がコグニマサーバから変更ログ項目を受信する場合：

- ・コグニマサーバから新規のオブジェクトメッセージを受信する場合、クライアント装置はこの新規のオブジェクトをローカルデータベースに追加する
- ・コグニマサーバからオブジェクト削除メッセージを受信する場合、クライアント装置はローカルデータベースにおいてこのオブジェクトに削除済としてマーク付けする
- ・クライアント装置がプロパティ変更を受信する場合、常に、コグニマサーバがデータベースの現在の状態に関して信頼できるものと想定されるので、変更はローカルデータベースに保持されるプロパティの値に行われる

コグニマサーバはクライアント装置から変更ログ項目を受信する：

- ・コグニマサーバがクライアント装置から新規のオブジェクトを受信する場合、オブジェクトはコグニマサーバのデータベースに追加され、最初に新規のオブジェクトを送信したマシンの変更ログを除いてそのユーザに対して登録されたクライアント装置の全ての変更ログにも追加される
- ・コグニマサーバがクライアント装置からオブジェクト削除を受信する場合、オブジェクトは削除のためにマーク付けされ、オブジェクト削除メッセージが、最初にオブジェクト

削除を送信したマシンの変更ログを除いてそのユーザに対して登録された装置の全ての変更ログに追加される

・プロパティ変更を受信する場合、コグニマサーバは変更の時間をコグニマサーバ上でそのプロパティに対して保持される現在時間と比較する。プロパティ変更の時間がコグニマサーバ上で保持される時間より後の場合、サーバデータベースにおいてプロパティ値が変更され、この変更はそのユーザに対して登録されたクライアント装置の全ての変更ログ・プロパティ変更において送信されるマシンのものを含む一にも追加される（別のオブジェクト更新がその間マシンに送信される場合）。プロパティ変更がコグニマサーバ上で保持されるものより前の場合、格納されたプロパティ値の方がより最近のものであるので変更は行われぬ。しかし、ユーザが必要に応じて後で取得できるように値はコグニマサーバ上の古いプロパティ値のリストに追加される。時間同士が比較されるときには、第6.2節にて後述する時間管理手法が使用される。

10

#### 【0059】

装置が最初にコグニマサーバへと接続する場合、全てのクラスメタデータ定義及びそのユーザに対するデータベース中の全てのオブジェクトが送信されることになる。一般的に、削除メッセージは削除のためにオブジェクトにマーク付けするだけである。オブジェクトのデータベースからの実際の削除は、そのオブジェクトを参照する全てのオブジェクトが削除された後で行われるであろう。

#### 【0060】

##### 5. 2. 5 最適化

20

上述の複製プロトコルの最適化バージョンは、変更ログ中のエントリの集約を考慮に入れている。変更ログ（コグニマサーバ中又はクライアント装置上）がまだ複製されていない場合、後続のエントリが追加される。既存のエントリを走査することにより、次の接続中に複製される必要のあるエントリ数を場合によっては削減することができる：

- ・新規のエントリが、更新が予定されているプロパティに対する更新である場合、後のエントリのみが保持されれば良い
- ・新規のエントリがオブジェクト削除である場合、そのオブジェクトに対する全てのプロパティ更新を変更ログから削除することができる
- ・新規のエントリが「復旧」コマンドであり、元の削除がまだ変更ログ内にある場合、2つのエントリは共に変更ログから削除することができる。

30

#### 【0061】

##### 6. コアアルゴリズム

##### 6. 1 エンディアンネスの処理

オペレーティングシステムは、基本的にリトルエンディアン及びビッグエンディアンのうちの何れかである。これは、数字及び文字が格納されるバイトの並び順の選択である。異なるエンディアンネスを有する2台のコンピュータが通信を行う必要がある場合、片方のコンピュータはデータパケットのエンディアンネスを切替えなければならないだろう。コグニマ環境では、コグニマクライアントソフトウェアはホストクライアント装置と同じエンディアンネスを使用する。コグニマサーバはクライアント装置のエンディアンネスを判定し（クライアントからのデータの最初のパケットに含まれる参照値を使用）、コグニマサーバにおいて一貫したエンディアンネスを維持する必要がある場合に後続の受信データを変換しなければならない。また、コグニマサーバはクライアント装置に送り返す送信データを変換しなければならない。

40

#### 【0062】

##### 6. 2 システム時間の同期化

異なる装置は必然的に多少異なるシステム時間を有することになる。クライアント装置からコグニマサーバへと送信される変更は、変更時の装置のシステム時間でスタンプされる。変更が発生した順序を判断し、正しい更新を記録することができるよう異なる装置上で時間を変換することはコグニマサーバの役目である。

#### 【0063】

50



装置のログオンは現在の装置の時間を含む。コグニマサーバはネットワークの待ち時間を補償し、ログイン時間をそれ自体のシステム時間と比較することができなければならない。これにより、装置の時間とコグニマサーバの時間との間に差分が生じることになる。この差分はそのセッションにおいて装置により送信される更なる時間に適用することができる。

#### 【0064】

コグニマサーバは、連続セッションにおける装置からの差分同士を比較し、装置上のクロック「クリープ」又は時間帯の変更を判定することができる。システム内の全てのクライアント装置が相互に十分に同期のとれたクロックを有することは以下に示すように想定しがたい：

- ・装置のクロック精度によってクロック時間は装置上で変動する
- ・ユーザの中には、例えば、5分早くクロックを設定することを好む人もいる
- ・ユーザの中には、ローカル設定を調整するよりはサマータイムのためにクロックを変更する人もいる（また、OSの中にはローカル特性を提供せず、ユーザがクロックを直接変更することを強制するものもあるかもしれない）

この問題を克服するために、サーバは、サーバ上で比較が行われる場合にクライアント装置により使用される時間をGMTへと調整し、メッセージがコグニマサーバからクライアント装置へと送信される場合にGMTからクライアント装置に対する同等の時間へと調整する役目を負うであろう。

#### 【0065】

クライアント装置は、変更ログ中の全ての項目にローカルクロックから取得された時間でタグ付けすることになる。クライアント装置に関する限りでは、クライアント装置自体のクロックに基づいて時間を扱う。

#### 【0066】

コグニマサーバへと接続するたびに、クライアント装置はそれ自体のクロックにより得られた現在時間のビューを送信する。これにより、サーバは以下のことを導き出すことができる：

- ・GMTに対する差分
- ・前回ログオンして以降モバイル装置のクロックに変動があった場合、サーバはGMTに対する最終差分及び最後の接続の時間を記録し、これらを比較することができる。変動がある場合、サーバはモバイル装置に応じて（pro-rata）送信される全ての時間を調整することができる。

#### 【0067】

例えば、以下の表に、コグニマサーバへと接続するクライアント装置におけるイベントのパターンを示す。クライアント装置の時間はコグニマサーバより5分遅く、毎時1分遅れている（要点を示すために極端な場合を挙げる）。更に要点を示すために、09：00から12：00までユーザは機上でコグニマサーバと接触がなく、この間クライアント装置は接続していないものとする：

アクション	クライアント装置の時間	コグニマサーバの時間 (GMT)
クライアント装置がコグニマサーバへと接続する	09：00	09：05
変更がプロパティAに加えられる	10：00	X
変更がプロパティBに加えられる	11：00	Y
クライアント装置がコグニマサーバへと接続する	12：00	12：08

プロパティ変更がコグニマサーバ上に格納された時間の前後に行われたか否かを導き出

10

20

30

40

50

すために、時間 X 及び Y を計算する必要がある。上述の情報から、コグニマサーバには、クライアントが前回接続したのは約 3 時間前であり、その時点では時間差は 5 分であったが、現在は 8 分になっていることが分かっている。クロック変動が直線的に発生するものと想定すると、コグニマサーバは、装置が GMT より 5 分遅れており、クロックは毎時 1 分遅れることを算出することができる。

#### 【0068】

これにより、クライアント装置がプロパティ A の変更に対して 10:00 として認識する時間は、初期変動として 5 分を加える必要があり、プロパティが変更されるまでの時間に発生した追加の変動のために更に 1 分加える必要がある。

#### 【0069】

同様に、プロパティ B は 11:07 に調整される必要がある - 5 分の初期変動に加えて、プロパティが変更されたとき 09:00 から 11:00 まで 2 時間経過しているために更に 2 分必要である。

#### 【0070】

実際には、クライアント装置の時間と GMT との間の時間に対する差分は数分であるが、変動は 1 時間あたり数分の 1 秒のオーダーとなるであろう。

#### 【0071】

##### 6. 2. 1 時間調整

GMT に対する差分及びクライアント装置のクロックと同様に、ユーザはクライアント装置上の時間を変更することができる。ユーザは、時間を正しい現地時間にリセットするためにこの変更を行うかもしれない（ユーザは自動的にこの変更を行うオプションを選択することもできるが、ユーザの中にはクライアント装置の時間を自分で制御したい - 例えば、クロックを 5 分早く設定したいと思う人もいるかもしれない）。また、現地時間（すなわち、サマータイム又は時間帯の変更）の変更を反映させるために変更を行うかもしれない。目的は、ユーザが自身に即した任意の時間に装置上のクロックを変更することができる、装置はただこれを考慮するだけであるように構成することである。

#### 【0072】

ユーザがクライアント装置の時間に対して変更を行う場合、大半のオペレーティングシステムはこの変更を報告する（これを行わないシステムにおいては、このような変更をチェックするために、例えば、毎分時間をポーリングすることができる）。時間の変更を検出すると、クライアント装置は新しい時間と変更前の時間との間の差分を算出するであろう。例えば、ユーザが時間帯を移動したときにはプラス 1 時間の変更となるかもしれない。クライアント装置は、この時間差を調整時間として格納する。この調整時間は、コグニマサーバへの次の接続時のために保存される。また、クライアント装置は、変更ログの全ての項目に目を通し、調整時間によりログ中の全ての時間を更新する。これにより、変更ログ中の各エントリをクライアント装置上の現地時間と常に関連させておくことができる。

#### 【0073】

このような調整は、コグニマサーバへの接続間でも行うことができるだろう - ログ中の各時間は全てクライアント装置上の現地時間に関連するように、毎回、時間変更の量が調整時間を用いてまとめられ、変更ログが更新される。

#### 【0074】

クライアント装置が次にコグニマサーバへと接続するときには、格納された調整時間 - すなわち、前回の接続以降クライアント装置のクロックが進められた又は遅らせられた量をログオン時に送信する。コグニマサーバは、GMT に対する差分及び変動計算により時間からこの量を差し引くことができる。

#### 【0075】

##### 6. 2. 2 クライアント装置に対する GMT

コグニマサーバ上で行われた変更の GMT 時間のあるクライアント装置にとって正しい現地時間に変更するべく、同じ一連の計算を逆に行うことができる。

10

20

30

40

50

## 【0076】

## 6.3 アプリケーションの追加

アプリケーションは1つ以上のクラスを使用してユーザデータを保持するであろう。クラスの定義はアプリケーションへとハードコードされる。クラスのバージョンは、アプリケーションのリリースにより調整される。

## 【0077】

統計アプリケーションがフットボール選手クラスを使用してフットボール選手に関するデータを保持するものとする。アプリケーションがクライアント装置上で最初に起動されるとき、どのバージョンのフットボール選手クラスを既に保持しているかを装置に問い合わせる。装置上のバージョンが、アプリケーションが使用すべくハードコードされたバージョンと同じである場合、それ以上何も行わなくて良い。

10

## 【0078】

装置が、フットボール選手クラスの新しいバージョンを保持する場合、アプリケーションは予想よりも多くのプロパティに対処できるよう十分頑強でなければならない（複数のアプリケーションにより使用されるクラスを有していて、何らかの理由でアプリケーションのうちの1つの古いバージョンをインストールした場合にこの状況が発生するであろう。これは稀なケースであり、理想的には相互依存するアプリケーションは共にアップグレードすべきである。）。

## 【0079】

装置がフットボール選手クラスの古いバージョンを保持する（又は全くバージョンを保持しない）場合、アプリケーションの有するフットボール選手クラスのバージョンをそれと置き換えるべきである。新しいバージョンはコグニマサーバへと送信される。コグニマサーバは全ての装置上で使用されるクラスの各バージョンのリストを維持する。

20

## 【0080】

ウェブポータルページは、ハードコードされた装置のアプリケーションと同等であろう。ウェブはクラスの最新バージョンに従ってデータベースからオブジェクトを抽出することができ、予想してハードコードされたよりも多くのプロパティがある場合は無視することができる。従って、ウェブはクラスバージョンを宣言する必要がない。

## 【0081】

## 6.4 変更ログの最適化

コグニマサーバは、装置が次に接続するときに送信されることになる変更を列挙する全ての装置に対する変更ログを維持する。例えば、変更ログに対して行うことができる最適化には以下に示すようなものがあるだろう：

30

- ・同じプロパティに対する3つ以上の更新が変更ログにキューイングされている場合、最後の更新のみを維持すれば良い
- ・あるオブジェクトに対して削除がキューイングされている場合、キューの前方にある更新は全て削除されても良い
- ・あるオブジェクトに対して更新がキューイングされている場合、キューの前方にある削除は削除されるべきである
- ・装置が新規のアプリケーションを登録する場合、そのアプリケーションに対して送信すべきオブジェクト（例えば、メッセージ履歴）が非常に多くなる可能性がある。変更ログは、追加対象として妥当な数のオブジェクト（例えば、最新の20通のメッセージ）のみを持つべきである

40

## 7. ゴースト化、復活 (resurrection)、ピン止め (pinning)、及び撤回 (withdrawal)

クライアント装置上で利用可能なユーザデータを保持する空間は、通常、サーバ上で利用可能な空間よりも数桁小さい。装置は、データのサブセットを保持する必要があり、ユーザはこのサブセットを維持するだけで良い。ゴースト化及び撤回はこれを支援するためのツールである。

## 【0082】

50

クラス定義は、ある特定のプロパティに「ゴースト化可能」としてフラグを立てることを含んでも良い。これは、オブジェクトがゴースト化される場合、プロパティがゼロになり、クライアント装置上の空間が解放されることを意味する。ゴースト化は装置上で自動的に行われる。どのオブジェクトをゴースト化すべきかの決定は、「ゴースト化規則」に従い、オブジェクトが作成又は更新されるたびに規則を適用することによって行われる。規則は、オブジェクトのセレクションの最大数を定義する。最大値を超える場合、ソート順の末尾にあるセレクション中のオブジェクトはゴースト化される。

#### 【0083】

例えば、クラスはメッセージであり、セレクションは受信ボックス内のメッセージであり、ソート順は日時であり、最大数は50であるかもしれない。受信ボックスに50通のメッセージがある状態で新規のメッセージが到着した場合、受信ボックス中の最も古いメッセージがゴースト化される。ゴースト化では、メッセージ本体を削除した場合でもメッセージを認識するのに十分なヘッダ情報を残しておいて良い。

10

#### 【0084】

撤回（従来は自動削除及び削除としても既知）は、ゴースト化に類似しているが、オブジェクトの一部ではなく全体を削除することによって機能する。

#### 【0085】

ゴースト化も撤回もコグニマサーバには通知されない。これらの操作はクライアント装置特有のものである。異なる装置は異なる個数のオブジェクトを有しても良い。装置上のデータは基本的に同期がとれているが、各装置は異なるデータ部分集合を保持する。

20

#### 【0086】

ユーザがゴーストの復活を希望する場合、オブジェクトが復活するようにクライアントからコグニマサーバへと要求が渡される。オブジェクトのリフレッシュが装置へと送信され、オブジェクトは正常に戻る。

#### 【0087】

個々のオブジェクトはピン止めすることができる。ピン止めされたオブジェクトは、ゴースト化されたり削除されたりすることはない。ピン止めはユーザが選択することも、自動的に行うこともできる。例えば、復活したオブジェクトは自動的にピン止めされる。

#### 【0088】

##### 8. ユーザ複製—オブジェクトの共有

30

ユーザがオブジェクトを共有するのに役立つと思われるアプリケーションは数多く存在する。これが有効に機能すると思われる一般的な場合は、ユーザがオブジェクトを共有したいユーザのコグニマアドレスを知る必要がある場合である。これらのアドレスの取得については、コグニマサーバアーキテクチャにおいて詳細に検討することがより適切である。ここでは、このようなリストが利用可能であると想定する。

#### 【0089】

1つ以上のコグニマアドレスの集合が共有対象のオブジェクトに添付される。オブジェクトは読出し専用を設定することができる（共有相手が修正できないように）。ウェブ又はクライアント装置から新規のオブジェクトを受信する（あるいは、オブジェクトに対する更新を受信する）と、コグニマサーバは通常通りそれを複製する。

40

#### 【0090】

また、コグニマサーバは「共有者」のコグニマアドレスのリストを参照する。オブジェクトに発信者のID（すなわち、オブジェクト所有者のコグニマアドレス+オブジェクトID）でマーク付けし、共有者へと送信する。共有ユーザは同じコグニマサーバ上又は異なるコグニマサーバ上に存在する。共有者のコグニマサーバがオブジェクトを受信する。新規のオブジェクトの場合には新規のオブジェクトIDを割り当てる（発信者IDを記録）。更新の場合には発信者IDを使用しているオブジェクトを探し出す。共有者がオブジェクトを更新する許可を受けている場合、発信者IDを使用して更新を逆にオブジェクト所有者へと複製することができる。

#### 【0091】

50

## 9. データの表示

PDAなどの従来の小型装置は単純なファイリングシステムを有する傾向にある。これらのシステムによりアプリケーションは、稼動中でないときにデータを保持する何らかの記憶装置においてデータを読み書きすることができる。一般的に、このようなプログラムは利用可能なデータの集合を読み込み、データを画面に表示するためのユーザインタフェースを提供する。これには幾つかの欠点がある：

- ・プログラムが起動するときのデータの読み込みに時間がかかる
- ・アプリケーションはデータの全部又は一部をメモリに格納する必要がある（つまり、クライアント装置上ではより多くのメモリを占めている）
- ・2つ以上のアプリケーションが同じデータの集合をアクセスできるようにすることは問題となる

・データを読んだり操作したりするための類似のコードが装置上で稼動する幾つかのアプリケーションで出現する

コグニマ手法はこれとは異なる：

- ・データは幾つかのアプリケーションによるアクセスが可能なオブジェクトデータベースに格納される
- ・コグニマアプリケーションは、データベースから扱う全てのデータを読み込まない。その代わりに、セレクションー現在関心のあるデータの部分集合ーを作成する。一般的に、このセレクションは装置の画面上に現在表示されているデータと一致する。このため、アプリケーションが現在使用中のデータのみがメモリに保持されるーメモリ空間を大いに節約することになる

・データの格納作業、ソーティング作業、及び索引付け作業の全てはオブジェクトデータベースにより行われるので、この機能を各アプリケーションにおいて繰り返す必要がない

・アプリケーションにおいてデータに変更を加える必要がある場合、アプリケーションはデータの表示画面を直接更新しない。変更は、オブジェクト中のプロパティを更新するか、あるいは、オブジェクトを作成又は削除することになる。データの変更は、別のアプリケーションにより、あるいは、別のマシン上でデータが変更されたためにコグニマサーバから受信された更新により行うことができるだろう

・アプリケーションがセレクションを設定するときには、データがセレクションに含まれるか、あるいは、除外されるかを定める基準のリストが与えられるーこれにより、コグニマ複製エンジンはオブジェクトが作成、削除、又は更新されたときにどのアプリケーションに通知すべきかが分かる

・更新をアプリケーションへと送信する必要がある場合、このデータを含むセレクションへとリンクされたアプリケーション中のコードが呼び出され、こうして、アプリケーションは行われた変更に応えることができる

・セレクションが設定されるとき、アプリケーションはデータのソート方法と必要なのはデータのソート済リスト上の小型のウィンドウのみであるか（ビューとして既知）とを指定することができる

この手法は、各ウィンドウイングシステム上でグラフィックスを再描画するのに使用される画面リペイント手法に類似している。画面のある領域がリペイントを必要とするとき、その画面の一部を担当するアプリケーションが画面のリペイントのために呼び出される。

### 【0092】

#### 9. 1 例

クライアント装置は、装置上で稼動する連絡先アプリケーションを有することもあるーこの装置は、同様に連絡先アプリケーションを実行中の他のクライアント装置に接続されるコグニマサーバを用いてデータを複製する。名前及び電話番号を含むある連絡先に対してオブジェクトのクラスが定義され、これらのデータが特定のユーザの全ての装置へと複製される。

### 【0093】

ある装置上のアプリケーションは、頭文字により全ての連絡先を表示する表示画面を有しても良い—例えば、インタフェースによりユーザがDのボタンを押下することで、Dで始まる全ての名前を表示することができる。このアプリケーションは、以下のような場合にオブジェクトを含むセレクションを設定するであろう：

- ・クラスが連絡先として定義される場合
- ・名前が選択された文字（例えば、D）で始まる場合

セレクションが定義される場合、アプリケーションはオブジェクトが追加、削除、又は更新されたときにC R Eにより呼び出されるコードも定義する。

#### 【0094】

セレクションが最初に設定される場合、このコードは上述の基準を満たす第1のオブジェクト集合を伴って呼び戻されることになる。 10

#### 【0095】

Dで始まる名前を有する新規の連絡先を作成するように要求された場合、アプリケーションはオブジェクトを作成する以外のことは行わない。C R Eは新規のオブジェクトを検出して、新規のオブジェクトを通知するためにセレクションコードを呼び戻すであろう。

#### 【0096】

同様の場合は、新規の連絡先オブジェクトが別の装置上で作成され、クライアント装置へと複製された場合である—連絡先の名前がDで始まる場合、アプリケーションは通知を受けるであろう。

#### 【0097】

### 9. 2 ソーティング 20

一般的に、セレクション中のデータはソートする必要がある—多くの場合、表示時にユーザが論理フォーマットでデータを見ることができるようにするためである。セレクションが定義されるときに、ソーティング順（どのプロパティによってソートすべきか、どの順序でソートすべきか、及びどのソーティングアルゴリズムを使用すべきか）を指定することができる。

#### 【0098】

### 9. 3 ビュー

セレクション中には多数のデータ項目が存在することもある。通常、データの表示中にデータが画面内に収まらない可能性もあり、ユーザはデータを上下にスクロールする必要があるだろう。ビューは、セレクションが扱うデータ項目の個数とアプリケーションがセレクション中に出したいデータを網羅するリストのうちの第1のデータ項目の個数とを指定することによってこの機能を提供する。 30

#### 【0099】

アプリケーションは、画面上に表示するのに必要な量のみ制限されるようにローカルで格納するデータ量を制限することができるので、ビューは重要である。これにより、データの不要な複製が減少する。

#### 【0100】

### 9. 4 効率

コグニマは、データがコグニマサーバとクライアントアプリケーションとの間で転送される方法において効率的な最適化を成し遂げた—複数のデータ変更が行われると、アプリケーションはユーザインタフェースを1回だけ更新すれば良いように、データはブロックで送信され、アプリケーションは変更が完了したことを通知される。 40

#### 【0101】

### 9. 5 例

一例として、ContactSelectionと呼ばれるセレクションを定義する。これは、選択されたオブジェクトのうちの何れかに対して変更が行われるたびにフレームワークが呼び戻すコードである。コグニマフレームワークでは、C0dbSelectテンプレート化クラス—テンプレート引数としてセレクション中に持たたいオブジェクトの型を指定—から取得するオブジェクトとして実現される。 50

```

class CContactSelect:public C0dbSelect<CContact>
{
public:
    CContactSelect(C0db *a0db);
    void ObjectAdded(CContact *aObject);
    void ObjectUpdated(CContact *aObject);
    void ObjectRemoved(const T0dbObjectId aObjectId);
private:
    bool ListContacts();
};

```

10

オブジェクトが追加、更新、又は削除されるたびにフレームワークにより各メソッド `ObjectAdded()`、`ObjectUpdated()`、及び `ObjectRemoved()` がそれぞれ呼び出される。セレクションクラスを実現する際に、これらのイベント発生時にインスタンスアクションを取りたくない場合には、全てのメソッドを実現する必要はない—ある特定のオブジェクト集合のリストを維持するためにセレクションを設定するが、他の何らかのイベント発生時にのみリストのチェックを行い、上述のメソッドが必要でない場合もある。

#### 【0102】

更にプライベートメソッド `ListContacts()` も定義している—これは、セレクションにより保持される全ての現在の連絡先を列挙する。

#### 【0103】

20

このクラスの実施例を挙げる：

```

CContactSelect::CContactSelect(C0db *a0db)
:C0dbSelect<CContact>(a0db)
{
}

void CContactSelect::ObjectAdded(CTestContact *aContact)
{
    OdbLog(OdbLogApp,L"New contact added:"<<aContact->GetName());
    ListContacts();
}

void CContactSelect::ObjectUpdated(CTestContact *aContact)
{
    OdbLog(OdbLogApp,L"Contact updated:"<<aContact->GetName());
    ListContacts();
}

void CContactSelect::ObjectRemoved(const T0dbObjectId aObjectId)
{
    OdbLog(OdbLogApp,L"Contact deleted - Id:"<<aObjectId);
    ListContacts();
}

void CContactSelect::ListContacts()
{
    OdbLog(OdbLogApp,L"Contacts list:");
    for(unsigned long Index=0;Index<iResult.size();Index++)
    {
        CTestContact *Contact=iResult[Index];
        OdbLog(OdbLogApp,Contact->GetName()<<"",
            <<Contact->GetPhone()<<"",
            <<Contact->GetEmail());
    }
}

```

30

40

50

}

コンストラクタは、デフォルトのC0dbSelectコンストラクタを呼び出すだけである。ObjectAdded()、Updated()、及びRemoved()メソッドは、何の変更があったのかをプリントアウトし、ListContacts()を呼び出してリストの現在の内容を示す。

【0104】

ListContacts()は、セレクションにより保持されるオブジェクトの現在のリストがどのようにアクセス可能になるかを示す。オブジェクトを指し示すポインタの現在のリストがiResultと呼ばれるコンテナクラスに保持される—これは、通常のコンテナクラスインテグレートによりアクセスすることができる。この場合、リストに目を通して、リスト中の全てのオブジェクトをプリントするだけである。

10

【手続補正書】

【提出日】平成16年3月10日(2004.3.10)

【手続補正1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項1】

複数のコンピュータ装置における第1のコンピュータ装置が、無線ネットワークを介して無線携帯コンピュータ装置から受信されたデータが複製されたか否かを判定する複数のコンピュータ装置からのデータを複製する方法であって、

(a) 変更されたレコードに対してのみデータを含む変更ログを生成する工程と、

(b) 前記第1のコンピュータ装置上で稼動するデータベースにおいて前記変更ログ中のデータを複製するために、前記第1のコンピュータ装置へその変更ログを送信する工程とを有し、

前記無線携帯コンピュータ装置が前記第1のコンピュータ装置を用いて新規のメタデータ定義を登録し、どのように新規のアプリケーションに関連する変更ログが前記無線コンピュータ装置により提示されるかを定義することを特徴とする方法。

【請求項2】

前記変更ログはレコードに対する変更が行われた時間を示すタイムスタンプを有し、前記時間は前記無線コンピュータ装置上のリアルタイムクロックを使用して測定されることを特徴とする請求項1記載の方法。

【請求項3】

前記変更ログは変更されたレコードのフィールドへの変更のみを記録し、前記レコードの全体を記録しないことを特徴とする請求項1記載の方法。

【請求項4】

前記無線コンピュータ装置と前記第1のコンピュータ装置との間で最後のデータ複製以来発生した変更のみを記録する工程を有することを特徴とする請求項1記載の方法。

【請求項5】

前記無線コンピュータ装置及び前記第1のコンピュータ装置が前記変更ログの適切な解釈を定義するメタデータを共有する工程を有することを特徴とする請求項1記載の方法。

【請求項6】

変更されたデータと、前記無線コンピュータ装置上でどのようにいつ変更されたかとを識別するデータが前記変更ログに入れられる工程を有することを特徴とする請求項1記載の方法。

【請求項7】

ユーザの介入なしに前記変更ログが前記無線コンピュータ装置によって前記第1のコンピュータ装置へ自動的に送信される工程を有することを特徴とする請求項1記載の方法。

【請求項8】



無線コンピュータ装置が、(a) 所定の定期的な間隔で、(b) 所定の時刻に、或いは(c) 変更ログが生成されたときはいつでも継続的に前記変更ログを自動的に送信する工程を有することを特徴とする請求項 1 記載の方法。

【請求項 9】

前記無線コンピュータ装置が複数の変更ログをまとめる工程を有することを特徴とする請求項 1 記載の方法。

【請求項 10】

前記無線コンピュータ装置が、ネットワークトラフィックがより低い時間に低い優先度の変更ログと共に、前記第 1 のコンピュータ装置で迅速に処理される変更ログの重要度に基づいた時間にバッチ・変更ログを送信する工程を有することを特徴とする請求項 9 記載の方法。

【請求項 11】

前記無線コンピュータ装置が、ネットワークトラフィックがより低い時間に低い優先度のレコードと共に、前記第 1 のコンピュータ装置で迅速に処理されるレコードの重要度に基づいた時間に変更ログ中のレコードを送信する工程を有することを特徴とする請求項 1 記載の方法。

【請求項 12】

第 1 のコンピュータ装置がデータを複製するために、無線ネットワークを介して前記第 1 のコンピュータ装置へ前記データを送信するようにプログラムされた無線携帯コンピュータ装置において、

(a) 変更されたレコードに対してのみデータを含む変更ログにおいてタイムスタンプを生成・格納し、

(b) 前記第 1 のコンピュータ装置上で稼動するデータベースにおいて前記変更ログ中のデータを複製するために、前記第 1 のコンピュータ装置へその変更ログを送信するようにプログラムされ、

前記第 1 のコンピュータ装置を用いて新規のメタデータ定義を登録し、どのように新規のアプリケーションに関連する変更ログが前記無線コンピュータ装置により提示されるかを定義するようにプログラムされた装置。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0006

【補正方法】変更

【補正の内容】

【0006】

【発明の概要】

本発明の第 1 の態様によると、複数のコンピュータ装置における第 1 のコンピュータ装置が、無線ネットワークを介して無線携帯コンピュータ装置から受信されたデータが複製されたか否かを判定する複数のコンピュータ装置からのデータを複製する方法であって、

(a) 変更されたレコードに対してのみデータを含む変更ログを生成する工程と、

(b) 前記第 1 のコンピュータ装置上で稼動するデータベースにおいて前記変更ログ中のデータを複製するために、前記第 1 のコンピュータ装置へその変更ログを送信する工程とを有し、

前記無線携帯コンピュータ装置が前記第 1 のコンピュータ装置を用いて新規のメタデータ定義を登録し、どのように新規のアプリケーションに関連する変更ログが前記無線コンピュータ装置により提示されるかを定義することを特徴とする方法が提供される。

【手続補正 3】

【補正対象書類名】明細書

【補正対象項目名】0007

【補正方法】変更

【補正の内容】

## 【0007】

この手法により、資源制約型無線携帯コンピュータ装置（例えば、携帯電話）は、低帯域幅の信頼度の低い無線接続を使用して過剰な処理負荷を負うことなくデータ複製機能（例えば、連絡先、電子メール、写真などのデータベースをリモート装置ーデータベースのマスタコピーを格納するサーバ又はピアーへとバックアップする）を提供できるようになる。

## 【手続補正4】

【補正対象書類名】明細書

【補正対象項目名】0008

【補正方法】変更

【補正の内容】

## 【0008】

機能は、変更されたデータベースから変更ログを切り離すことによってこれを実施する。リモート装置へと送信すべきデータは格段に少なくなり、通常、変更されたフィールドとそのフィールドが無線コンピュータ装置上で変更された方法及び日時だけである。従来のシステムでは、変更されていないデータを含んでいる時でさえ、通常、レコード全体を送信する。

更に、前記無線携帯コンピューティング装置は、前記第1のコンピューティング装置を用いて新規のメタデータ定義を登録し、どのように新規のアプリケーションに関連する変更ログが前記無線コンピューティング装置により提示されるかを定義するので、各変更ログは自己記述される（XMLベースのシステムがそうであるように）必要がなく、変更ログは実質的に生データとすることができる。このため、所要帯域幅を最小限にすることができる。

更に、変更ログは変更時間を示すタイムスタンプを含んでも良く、これにより、データベース自体にタイムスタンプを含む必要がなくなる。この手法では、従来行われているように各データベースレコードにタイムスタンプを押す必要がないので、無線装置上で相当なメモリ空間を節約することになる。通常、変更ログは、無線コンピュータ装置と第1のコンピュータ装置との間の最後のデータ複製以降発生した変更のみを記録する。従来のシステムでは、同期化より前のタイムスタンプを記録することによって貴重なメモリを浪費することが多い。

## 【手続補正5】

【補正対象書類名】明細書

【補正対象項目名】0011

【補正方法】変更

【補正の内容】

## 【0011】

第2の態様において、第1のコンピュータ装置がデータを複製するために、無線ネットワークを介して前記第1のコンピュータ装置へ前記データを送信するようにプログラムされた無線携帯コンピュータ装置において、

(a) 変更されたレコードに対してのみデータを含む変更ログにおいてタイムスタンプを生成・格納し、

(b) 前記第1のコンピュータ装置上で稼動するデータベースにおいて前記変更ログ中のデータを複製するために、前記第1のコンピュータ装置へその変更ログを送信するようにプログラムされ、

前記第1のコンピュータ装置を用いて新規のメタデータ定義を登録し、どのように新規のアプリケーションに関連する変更ログが前記無線コンピュータ装置により提示されるかを定義するようにプログラムされた装置が提供される。

更なる詳細は添付の請求の範囲において定義される。

## 【国際調査報告】

INTERNATIONAL SEARCH REPORT		Internat'l Application No. PCT/GB 02/05308
A. CLASSIFICATION OF SUBJECT MATTER IPC 7 606F17/30		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) IPC 7 606F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, INSPEC		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X A	US 5 884 325 A (BODGE ANDREW ET AL) 16 March 1999 (1999-03-16) column 1, line 53-55 column 2, line 32-39 column 8, line 65 -column 9, line 14 column 9, line 30-36 column 9, line 43-47 ----- -/-	1-5,7-9, 12 6
<input checked="" type="checkbox"/> Further documents are listed in the continuation of box C. <input checked="" type="checkbox"/> Patent family members are listed in annex.		
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier document but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art. "&" document member of the same patent family		
Date of the actual completion of the international search 19 December 2003		Date of mailing of the international search report 05/01/2004
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016		Authorized officer Correia Martins, F

## INTERNATIONAL SEARCH REPORT

 Internat  
 PCT/GB 02/05308

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	PACITTI E ET AL: "Improving data freshness in lazy master schemes" DISTRIBUTED COMPUTING SYSTEMS, 1998. PROCEEDINGS. 18TH INTERNATIONAL CONFERENCE ON AMSTERDAM, NETHERLANDS 26-29 MAY 1998, LOS ALAMITOS, CA, USA, IEEE COMPUT. SOC, US, 26 May 1998 (1998-05-26), pages 164-171, XP010283984	1-5,7-9, 12
A	ISBN: D-8186-8292-2 page 166, right-hand column, line 39-43	6
A	GRAY J METZ S: "SOLVING THE PROBLEMS OF DISTRIBUTED DATABASES" DATA COMMUNICATIONS, MCGRAW HILL. NEW YORK, US, vol. 12, no. 10, 1 October 1983 (1983-10-01), pages 183-192, XP000744182 ISSN: 0363-6399 page 187, paragraph UPDATING	7,8
X	ROBERT K LOCKHART ET AL: "Specifications for Ir Mobile Communications" SPECIFICATIONS FOR IR MOBILE COMMUNICATIONS (IRMC), XX, XX, 1 March 1999 (1999-03-01), XP002155443	1-5,7-9, 12
A	page 30, paragraph 5.5	6

## INTERNATIONAL SEARCH REPORT

International Application No. PCT/GB 02/05308

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

Continuation of Box I.2

Claims Nos.: 10,11

The application does not meet the requirements of Article 5 PCT because claim 10 claims the step in which batched change logs are sent at times which depend on the importance of the change log being rapidly processed by the first computing device.

The description does, however, not provide any information about how the "importance" should be assessed.

Therefore, the application as a whole does not sufficiently disclose the invention in a manner sufficiently clear and complete for it to be carried out by a person skilled in the art, Art. 5, PCT.

The same objection applies to claim 11 for similar reasons.

The applicant's attention is drawn to the fact that claims, or parts of claims, relating to inventions in respect of which no international search report has been established need not be the subject of an international preliminary examination (Rule 56.1(e) PCT). The applicant is advised that the EPO policy when acting as an International Preliminary Examining Authority is normally not to carry out a preliminary examination on matter which has not been searched. This is the case irrespective of whether or not the claims are amended following receipt of the search report or during any Chapter II procedure.

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/GB 02/05308

**Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)**

This International Search Report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:  
because they relate to subject matter not required to be searched by this Authority, namely:
  
2. ☒ Claims Nos.: 10, 11  
because they relate to parts of the International Application that do not comply with the prescribed requirements to such an extent that no meaningful International Search can be carried out, specifically:  
see FURTHER INFORMATION sheet PCT/ISA/210
  
3. ☐ Claims Nos.:  
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

**Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)**

This International Searching Authority found multiple inventions in this International application, as follows:

1. ☐ As all required additional search fees were timely paid by the applicant, this International Search Report covers all searchable claims.
  
2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
  
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this International Search Report covers only those claims for which fees were paid, specifically claims Nos.:
  
4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this International Search Report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

**Remark on Protest**

- ☐ The additional search fees were accompanied by the applicant's protest.
- ☐ No protest accompanied the payment of additional search fees.

## INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No  
PCT/GB 02/05308

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5884325	A	16-03-1999	NONE

## フロントページの続き

(81)指定国 AP(GH,GM,KE,LS,MW,MZ,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT, BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,IE,IT,LU,MC,NL,PT,SE,SK,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW, ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DZ,EC,EE,ES, FI,GB,GD,GE,GH,GM,HR,HU,ID,IL,IN,IS,JP,KE,KG,KP,KR,KZ,LC,LK,LR,LS,LT,LU,LV,MA,MD,MG,MK,MN,MW,MX,MZ,N O,NZ,OM,PH,PL,PT,RO,RU,SD,SE,SG,SI,SK,SL,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VN,YU,ZA,ZM,ZW

(特許庁注：以下のものは登録商標)

U N I X

J A V A

(74)代理人 100115071

弁理士 大塚 康弘

(74)代理人 100116894

弁理士 木村 秀二

(74)代理人 100101063

弁理士 松丸 秀和

(74)代理人 100130409

弁理士 下山 治

(72)発明者 イースト, サイモン, ジェレミー

イギリス国 ロンドン NW1 9JS, ウィルモット プレイス 3ビー

Fターム(参考) 5B082 AA04 GA04 HA03